Pandora: A Tool to Estimate Dimensionality Reduction Stability of Genotype Data

Julia Haag[2], Alexander I. Jordan[3], and Alexandros Stamatakis[1,2,4]

[1]Biodiversity Computing Group, Institute of Computer Science, Foundation for Research and Technology - Hellas, Heraklion, Crete, Greece
[2]Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Heidelberg, Germany
[3]Computational Statistics Group, Heidelberg Institute for Theoretical Studies, Heidelberg, Germany
[4]Institute for Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany

March 2024

Abstract

Motivation: Genotype datasets typically contain a large number of single nucleotide polymorphisms for a comparatively small number of individuals. To identify similarities between individuals and to infer an individual's origin or membership to a cultural group, dimensionality reduction techniques are routinely deployed. However, inherent (technical) difficulties such as missing or noisy data need to be accounted for when analyzing a lower dimensional representation of genotype data, and the uncertainty of such an analysis should be reported in all studies. However, to date, there exists no stability estimation technique for genotype data that can estimate this uncertainty.

Results: Here, we present Pandora, a stability estimation framework for genotype data based on bootstrapping. Pandora computes an overall score to quantify the stability of the entire embedding, per-individual support values, and deploys a k-means clustering approach to assess the uncertainty of assignments to potential cultural groups. In addition to this bootstrap-based stability estimation, Pandora offers a sliding-window stability estimation for whole-genome data. Using published empirical and simulated datasets, we demonstrate the usage and utility of Pandora for studies that rely on dimensionality reduction techniques.

Data and Code Availability: Pandora is available on GitHub `https://github.com/tschuelia/Pandora`. All Python scripts and data to reproduce our results are available on GitHub `https://github.com/tschuelia/PandoraPaper`.

Contact: `julia.haag@h-its.org`

1 Introduction

Dimensionality reduction techniques such as Principal Component Analysis (PCA) and Multidimensional Scaling (MDS) are popular in a plethora of scientific fields as they facilitate data visualization and interpretability. Both methods project high dimensional data to lower dimensions, while preserving as much variation, and thus information, as possible. Ever since the first application of PCA to genetic data almost half a century ago [16], PCA as well as alternative dimensionality reduction techniques have been widely used in modern population genomic analyses, including ancient DNA studies, to draw conclusions about population structure, genetic variation, or demographic history. For example, based on PCA results, Hughey et al. [13] found that the Minoan civilization did not originate from Africa. However, due to common challenges such as missing data and noise, the deployment of dimensionality reduction techniques to analyze ancient DNA or conduct contemporary population genetics studies is controversial. For instance, Yi and Latch [28] demonstrated that missing data can bias PCA analyses. Thus, quantifying uncertainty is important to properly and cautiously interpret results. For example, the relative location of populations/individuals in a PCA/MDS embedding determines its/their relation to other populations or individuals. If, however, there exists a substantial uncertainty with respect to the projection location for some or even all individuals in a study, the respective conclusions may potentially be erroneous and hence misleading. To the best of our knowledge, there currently exists no state-of-the art method to quantify the inherent uncertainty of dimensionality reduction techniques in ancient or contemporary population genetic studies. To address this challenge, we introduce Pandora, an open-source tool that estimates the uncertainty of PCA and MDS analyses for population genetics and ancient DNA genotype datasets. Pandora estimates the uncertainty via bootstrapping. In addition to an overall, global stability score, Pandora also estimates the stability of a subsequent k-means clustering for the computed embeddings. Both stability scores have values that range between zero and one. Higher values indicate a higher stability and thus a lower uncertainty with respect to the computed embedding for the respective dataset. Beyond this, Pandora also computes bootstrap support values for all individuals in the analyzed dataset. These support values indicate whether an individual's location in the lower-dimensional embedding space is stable across multiple bootstrap embeddings, or if the individual should be considered as being "rogue" when its projections between distinct bootstrapped embeddings differ substantially. For such a "rogue" individual, any conclusion concerning, for instance, the assignment to a specific (sub-)population should be carefully (re-)considered.

Estimating the stability of PCA using bootstrapping is not a novel concept. For example, Fisher et al. [10] propose a bootstrap-based method to estimate the sampling variability of PCA results for datasets with a substantially larger number of measurements per sample than samples itself. However, the standard bootstrapping methods are not directly applicable to genotype data, as we will discuss in Section 2.

In addition to bootstrap-based stability analyses, Pandora can also estimate the stability of dimensionality reduction in the context of whole genome analyses. Via a sliding-window approach, Pandora initially applies dimensionality reduction on subsequent equally sized subsets (windows) along the genome and then compares all pairs of induced per-window embeddings to each other. This allows to quantify the differences among embeddings in different parts of the genome.

Pandora is available on GitHub `https://github.com/tschuelia/Pandora`. It can be used via the command line and as a Python library. Pandora's command line interface requires the input dataset in either of the common genotype data formats. Pandora supports the three file formats defined by the EIGENSOFT software package [18] (*EIGENSTRAT*, *ANCESTRYMAP*, *PACKEDANCESTRYMAP*), as well as the *pedigree*, and *binary pedigree* formats defined by the PLINK software package [22]. When using Pandora as Python library, users can analyze an arbitrary numerical data matrix of biological data other than genotype data (e.g., gene expression data) provided as a *NumPy* [12] array. A thorough doc-

umentation of the tool, including usage examples, is available at `https://pandorageno.readthedocs.io`.

In Section 2 we provide a detailed description of the stability estimation as implemented in Pandora. In Section 3 we demonstrate its utility by using and re-analyzing published empirical as well as simulated population genetics datasets. We conclude in Section 4.

2 Materials and Methods

Figure 1 depicts our algorithm to estimate the stability of a genotype dataset under dimensionality reduction using bootstrapping. We first generate N bootstrap replicates $B^1, \ldots, B^N$ by randomly sampling from the set of Single Nucleotide Polymorphisms (SNPs) in the input genotype data with replacement. For each of the bootstrap replicates B^i, we perform a dimensionality reduction using either PCA or MDS depending on the user's choice. We denote the resulting embedding of bootstrap replicate B^u as P^u. To estimate the stability under bootstrapping and to compute support values for each individual, we conduct a pairwise comparison of all bootstrap embeddings $P^1 \ldots P^N$. To compare two distinct bootstrap embeddings P^u and P^v where $u \neq v$, we use a Procrustes Analysis, which finds a transformation that maps P^u as accurately as possible to P^v. An important property of the resulting transformation is that the relative distances between individual points in the embedding space are preserved. Based on these transformed projections, we can then compute the overall Pandora Stability (PS), as well as the Pandora Support Values (PSVs) for all individuals included in the dataset. We additionally perform k-means clustering for each bootstrap embedding and compare the resulting cluster label assignments to each other, to calculate the Pandora Cluster Stability (PCS).

An analogous graphical overview for the Pandora sliding-window analyses is provided in Supplementary Figure S1.

In the following, we briefly explain the fundamentals of PCA and MDS. We also provide information on the respective implementations we used in Pandora. We further describe Pandora's bootstrap procedure in more detail, with a focus on the pairwise comparison of two embeddings, as well as the computation of the PS, PSVs, and the PCS. Finally, we briefly explain Pandora's sliding-window analyses for whole genome data.

Henceforth, the term 'population' refers to its biological rather than to its statistical interpretation. We refer to a genotype data sequence in the input data as a single individual and, in general, each individual is associated to a population (e.g., "Greek").

2.1 PCA and MDS

PCA and MDS are statistical methods to reduce the dimensionality in data that exhibit a high number of observations/features per individual. The goal is to reduce the dimensionality while preserving as much information as possible, in order to facilitate the interpretability and visualization of high-dimensional input data.

PCA transforms the input feature space into a new, lower dimensional embedding space by projecting the data along the axes explaining the most variance in the data. These axes are called Principal components (PCs). The first PC explains the most variance, the second PC explains the second most variance, etc. How many PCs one should compute constitutes a challenging question. Patterson et al. [18] propose to base this decision on the Tracy-Widom statistic, but Elhaik [7] claim that this statistic is sensitive and results in overestimating the appropriate number of PCs. Applying threshold-based heuristics, for example, computing as many PCs as needed to explain 80% of variance, a criterion commonly used in other PCA application domains is not applicable to genotype data due to its extremely high dimensionality. For example, to explain 80% of the variance for a dataset analyzed by Lazaridis et al. [14], over 500 PCs need to be computed. In analogy to Price et al. [21], we thus set the default number of PCs in Pandora to 10, but allow users to specify any number of PCs.

Pandora supports two distinct input data representations. Using Pandora's command line interface requires users to provide the input data in one of

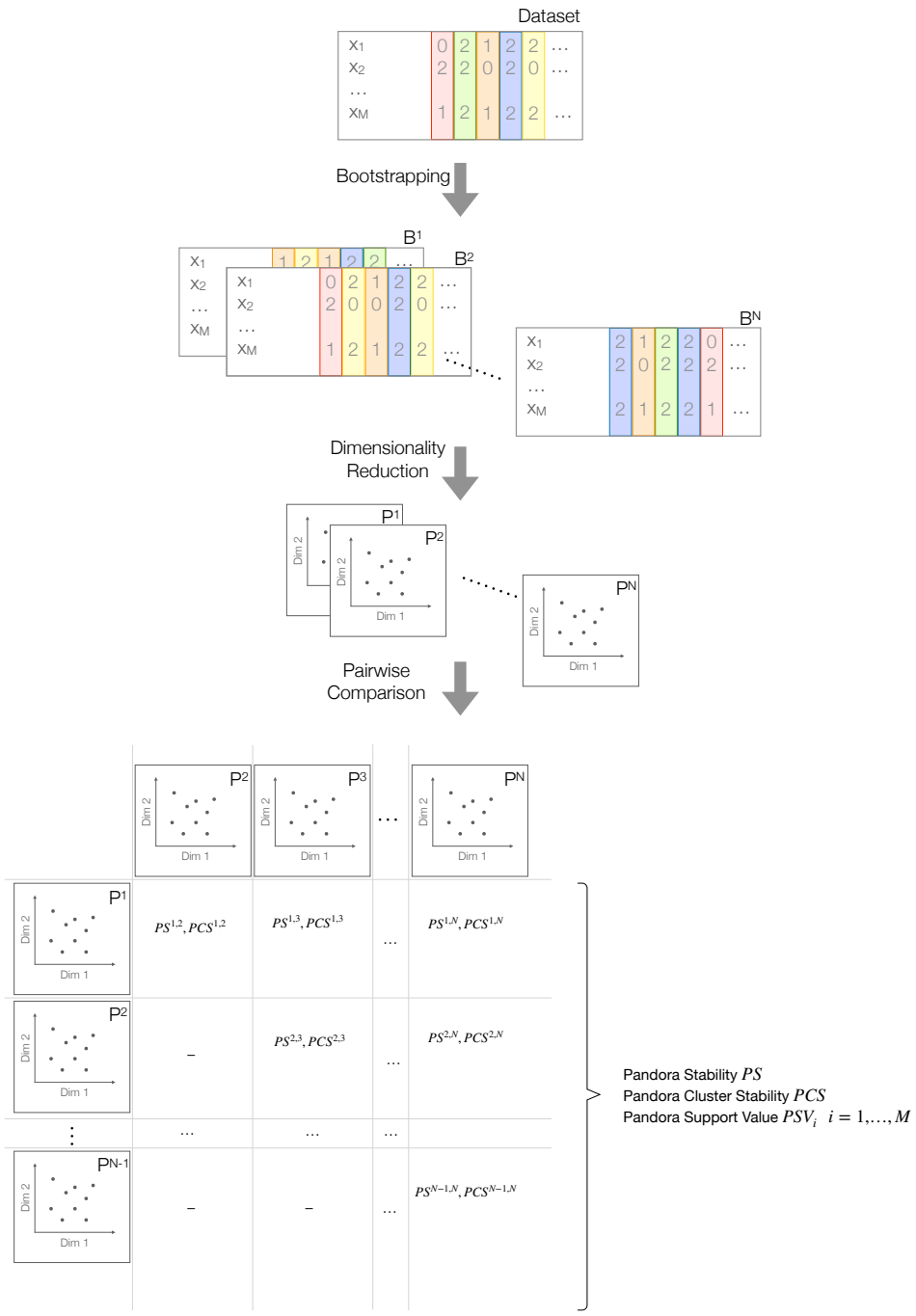


Figure 1: Schematic overview of the bootstrap-based stability analyses for genotype data, as implemented in our Pandora tool.

the common genotype file formats. Pandora supports the three file formats defined by the EIGENSOFT software package [18] (*EIGENSTRAT*, *ANCESTRYMAP*, *PACKEDANCESTRYMAP*), as well as the *pedigree*, and *binary pedigree* formats defined by the PLINK software package [22]. For this file-based command line interface, Pandora expects three distinct files per dataset: one genotype file containing the sequence data, one file containing metadata for each SNP, as well as one file containing metadata for each individual. For these file-based analyses, Pandora uses EIGENSOFT's *smartpca* tool, as it has been specifically designed and highly optimized for population genetics data [18, 21]. For PCA analyses, Pandora directly applies *smartpca* to compute all embeddings. As a consequence, Pandora supports and propagates all additional *smartpca* PCA analysis commands, including the option to project (ancient) individuals onto an embedding computed using another, distinct set of individuals.

The alternative dataset representation is based on *NumPy* matrices [12]. This enables users to preprocess their dataset using custom (Python) scripts without requiring an explicit (and potentially complicated) conversion into a specific population genetics file format. When using this alternative representation, Pandora performs all PCA analyses using the PCA implementation from the Python-based *scikit-learn* [20] machine learning library. Hence, this input file format allows for a more generic deployment of Pandora in other fields of biology beyond population genetics, for example for PCA analyses of gene expression data.

In contrast to PCA, MDS is not directly applied to the genotype data. Instead, MDS uses a distance matrix as input and reduces the dimensionality of the data while preserving pairwise distances as accurately as possible. Since genotype data can be represented numerically, we compute a metric MDS. With metric MDS, users can in principle choose any type of pairwise distance metric to analyze their genotype data. This distance metric can be computed either on a per-individual or per-population basis. A commonly used distance metric is the genetic F_{ST} distance between populations [27] which reflects the proportion of total heterozygosity that can be explained by the within-population heterozygosity. For the file-based command line interface, Pandora computes the F_{ST} distance matrix using *smartpca* and subsequently applies metric MDS as implemented in *scikit-learn*. When using the alternative *NumPy* based dataset representation, Pandora computes the F_{ST} matrix by default using the respective Python module *scikit-allel* [17]. However, Pandora also provides alternative distance metrics such as the Euclidean, Manhattan, or Hamming distances. Again, Pandora performs the subsequent MDS analysis using the *scikit-learn* implementation.

To keep the following description of Pandora as generic as possible, we henceforth refer to the result of a PCA or MDS analysis as *embedding*.

2.2 Genotype Data and Bootstrapping

Pandora was specifically designed for stability analyses of dimensionality reduction on genotype data. This genotype data consists of sequences of SNPs, that is, the genetic variation between individuals with respect to their DNA. Note that Pandora, in analogy to the EIGENSOFT [18] file specification, assumes that data are biallelic. In this case, a specific locus in the genome can be either one of two alleles a or A. Consequently, the two base pairs in the complementary DNA strands can take one of four states: aa, aA, Aa, or AA. This genetic information is encoded for each SNP and each individual under study. Note that the states aA and Aa are typically encoded using the same character, since the resulting genotype is independent of the order of the alleles. Finally, a dedicated special character denotes missing genotypes.

In the following, we represent the genotype data as a matrix $G \in \mathbb{R}^{M \times K}$, with rows corresponding to sequences (individuals) $x_1, x_2, \ldots, x_M$ and columns corresponding to SNPs $s_1, s_2, \ldots s_K$. Note that typically $M \ll K$ for genotype data.

The bootstrap procedure [6] is a statistical approach to quantify measures of accuracy based on random sampling with replacement from a set of observations. In statistics, the bootstrap is typically applied by sampling the individuals x_i. However, in

our case, instead of sampling the individuals, we sample the SNPs s_i to obtain a new bootstrap replicate matrix $B \in \mathbb{R}^{M \times K}$. Thus, a bootstrap replicate has the exact same set of individuals but with a distinct SNP composition. In doing so, we follow the standard approach used in phylogenetics and first proposed by Felsenstein [9]. A necessary condition for bootstrapping is the independence of the bootstrapped measurements. Individuals (especially individuals of the same species) do, however, in general not evolve independently of each other, but Felsenstein [9] argues that we may assume an independent evolution of genetic loci.

The computation of embeddings for individual bootstrap replicates are independent of each other, and their computation is therefore trivial to parallelize. Pandora implements this parallel computation and allows users to specify the number of threads to use.

2.3 Comparing Embeddings

In order to compare two embeddings P^u and P^v, we need to find the optimal transformation that matches the projections of all individuals x_i in P^u and P^v as accurately as possible while preserving the relative distances between individuals. Thus, the goal is to find a transformation f, such that

$$f(P^u) = P^v \quad \text{with} \quad f(P^u) = pA^T P^u + b. \tag{1}$$

The transformation consists of a scaling factor p, a rotation and reflection matrix A, and a translation matrix b. This equation can be solved using Procrustes Analysis. In Pandora, we use the Procrustes Analysis as implemented in the Python scientific computing package *SciPy* [25]. Prior to computing the optimal transformation, both embeddings are standardized such that $trace(PP^T) = 1$, and both sets of points are centered around the origin. The optimal transformation is then computed by minimizing the squared Frobenius norm (also called disparity)

$$D^{u,v} = \|f(P^u) - P^v\|_F^2. \tag{2}$$

The disparity $D^{u,v}$ has a minimum of 0 and a maximum of 1 [26]. Based on this pairwise matching of embeddings, we can now compute the three components of the Pandora stability analysis: the overall Pandora Stability (PS), the Pandora Cluster Stability (PCS) based on k-means clustering, and the per-individual Pandora Support Values (PSVs).

2.3.1 Pandora Stability

The Pandora Stability (PS) describes the overall stability of the genotype dataset based on the pairwise similarity scores over all bootstrap replicates. To compute the similarity between two bootstrap embeddings P^u and P^v we adopt the approach suggested by Wang et al. [26] and compute

$$PS^{u,v} = \sqrt{1 - D^{u,v}}. \tag{3}$$

The Procrustes analysis requires P^u and P^v to be of identical dimensions. Since we compute the same number of PCs for all bootstrap replicates, one dimension is always identical. However, by default, *smartpca* performs five rounds of outlier detection and removal that might result in a varying number of individuals per bootstrap. Thus, to ensure comparability, we only consider the projections of those individuals that are present in both bootstrap embeddings.

To obtain the overall PS, we simply average over the pairwise similarity across all $\frac{N(N-1)}{2}$ possible distinct pairs of bootstrap replicates

$$PS = \frac{2}{N(N-1)} \sum_{\substack{1 \leq u,v \leq N \\ u<v}} PS^{u,v}. \tag{4}$$

The resulting PS is a value between 0 and 1. The higher the PS, the more similar the bootstrap replicates are, with 1 indicating that all bootstrap embeddings project all individuals identically. In general, the higher the PS, the more stable the analyzed genotype dataset is under bootstrapping.

2.3.2 Pandora Cluster Stability

In analogy to the PS, we first compute the pairwise Pandora Cluster Stability (PCS) for all unique pairs

of bootstrap embeddings and then average over all pairwise scores to obtain the overall PCS

$$PCS = \frac{2}{N(N-1)} \sum_{\substack{1 \leq u,v \leq N \\ u<v}} PCS^{u,v}. \quad (5)$$

To compute the pairwise $PCS^{u,v}$ for bootstrap embeddings P^u and P^v, we first match both embeddings using Procrustes Analysis as described above and subsequently perform an independent k-means clustering for both embeddings. In order to compare the resulting label assignments for all individuals, we use the same number of clusters k for both embeddings. Finally, we compute the pairwise $PCS^{u,v}$ for bootstrap embeddings P^u and P^v via the Fowlkes-Mallows index (FMI) [11]

$$PCS^{u,v} = FMI = \frac{TP}{\sqrt{(TP+FP)(TP+FN)}}, \quad (6)$$

where TP denotes the true positives, FP the false positives, and FN the false negatives. Note that the computation of the FMI requires one set of cluster labels to be defined as ground truth labels. However, the FMI score is identical for either ground truth choice, since choosing the other set as ground truth only flips the FP and FN counts, which then only flips the terms in the denominator of Equation (6). The PCS is a value between 0 and 1. If the PCS is 0, the cluster assignments across all bootstraps and individuals are entirely different. If the PCS is 1, the cluster label for each individual is identical across all bootstraps. The higher the PCS, the higher is the similarity between clustering results across bootstraps.

Selecting the number of clusters k to best represent the dataset is not trivial. Pandora users can manually set the number of clusters based on, for instance, prior knowledge of the underlying (sub-)population structure of the data. In case the user does not manually set k, we automatically determine the optimal k based on a grid search and the Bayesian Information Criterion (BIC) [24]. To reduce the computational cost of searching for the optimal k, we determine an input data specific upper bound via the following heuristic. If the individuals of the dataset to be analyzed have population annotations, we set the upper bound to the number of distinct populations in the dataset. If this information is not available, we set the upper bound to the square root of the number of individuals in the dataset. Based on empirical parameter exploration experiments, we set the lower bound to 3. During Pandora development, we observed a substantial influence of k on the resulting PCS scores for various empirical population genetics datasets. We provide further details on these experiments in Section 3.

2.3.3 Pandora Support Values

To assess the projection stability for each individual in the dataset, we again rely upon the pairwise comparison between bootstrap embeddings as described in the previous section. We compute the Pandora support for each individual x_i as

$$PSV_i = 1 - \frac{\sum_{\substack{1 \leq u,v \leq N \\ u<v}} \|x_i^u - x_i^v\|}{2N \cdot \sum_{1 \leq u \leq N} \|x_i^u\|}. \quad (7)$$

The fraction in Equation (7) corresponds to the Gini coefficient, a statistical measure of dispersion from economics. In our case, the Gini coefficient measures the dispersion of the projections of individual x_i with respect to all bootstrap replicates on a scale from 0 to 1. The higher the dispersion, the higher the Gini coefficient will be. To align with the interpretation of the PS and PCS values, where 0 corresponds to unstable and 1 to stable, we simply subtract the calculated dispersion measure from 1.

2.4 Bootstrap Convergence Criterion

Since Pandora bootstraps an embedding (a compute-intensive dimensionality reduction), the resource usage and compute time is substantial. To reduce this computational overhead, we periodically perform a heuristic convergence assessment until a maximum number of bootstrap replicates is reached. The frequency of this convergence check is determined by the number of threads used.

The default setting is a maximum of 100 bootstrap replicates. Now, let N^* be the number of boot-

strap replicates already computed when performing the convergence assessment. We create 10 subsets of size $\tilde{N} = \frac{N^*}{2}$ by sampling 10 times without replacement from the set of N^* completed bootstraps. We then compute the Pandora Stability (PS) for each of the 10 subsets and determine the relative difference of PS values between all pairs of PS values (PS_u, PS_v): $\frac{|PS_u - PS_v|}{PS_u}$. We assume convergence if no pairwise relative difference exceeds α. In other words, we require that the highest sampled stability is at most $\alpha * 100\%$ larger than the lowest sampled stability. If this is the case, all remaining bootstrap computations are cancelled. Pandora allows users to explicitly set the convergence tolerance α; the default setting is 0.05. Note that we decided to sample 10 subsets of size $\tilde{N}$ as a trade-off between the accuracy and the additional runtime overhead of the bootstrap convergence assessment. The convergence assessment runtime overhead is substantial, as we need to compute the PS for each of the subsets, meaning that we need to perform $\frac{\tilde{N}(\tilde{N}-1)}{2}$ Procrustes Transformations.
We also determine the frequency of this convergence assessment based on the number of threads specified by the user, which defines the size of a full batch of bootstrap replicates. Pandora checks for convergence after every full batch or every 10th replicate, whichever is larger. For example, if a user executes Pandora with 40 threads, it will compute 40 bootstrap replicates in parallel and hence complete 40 replicates at approximately the same time. If Pandora were to run the convergence assessment after completing 10 bootstraps and determined that it has converged, we would discard 30 (almost) finished bootstraps, thus wasting the resources used for their computation, as well as for three unnecessary convergence assessment computations (after 10, 20, and 30 completed replicates). As a consequence of this thread-dependent convergence check, the exact results of the Pandora stability analysis may deviate between runs using different numbers of threads. However, as we demonstrate in Section 3 below, this deviation is only minor.

2.5 Whole Genome Sliding-Window Analysis

In addition to the above bootstrap stability analyses for genotype data, Pandora also implements a sliding-window stability analysis for whole genome data. The idea is to quantify differences in dimensionality reduction based observations for different subsets/parts of the entire genome. Instead of drawing and comparing bootstrap samples, Pandora splits the input dataset into N overlapping, consecutive subsets (windows) $W^1, \ldots, W^N$. In analogy to the bootstrapping procedure, the default setting is $N := 100$.

Let K be the number of SNPs in the dataset. Pandora splits the dataset into windows, such that the overlap between two consecutive windows is $\frac{K}{2N}$ SNPs and the offset between window frames is $\frac{K}{N}$. Thus, each window consists of $\frac{K}{N} + \frac{K}{2N}$ SNPs. Note that the final window will contain fewer SNPs as there is no consecutive window to overlap with. Then, Pandora applies dimensionality reduction to each window W^u to obtain the respective embedding P^u, and subsequently compares all resulting embeddings $P^1, \ldots, P^N$ to obtain the PS, PCS, and PSV scores. The comparison of embeddings, and thus the computation of the stability estimates, is identical to the bootstrap-based procedure outlined above.

3 Results

To demonstrate the utility of Pandora, we analyzed 19 empirical and 16 simulated genotype datasets obtained from various publications. In many publications on human sequences, especially in studies publishing newly sequenced ancient specimen, the Human Origins (HO) SNP Array [14] is used to compute the PCA. Ancient individuals are subsequently projected onto the resulting PCA. Based on the paper by Lazaridis et al. [14], we assembled the West-Eurasian subset using the 2068 publicly available individuals. The resulting West-Eurasian subset contains 813 individuals from 55 distinct populations. We further denote this dataset as *HO-WE*. Additionally, we merged the *HO-WE* dataset with the 13 ancient Çayönü individuals published by Altınışık

et al. [1] (*HO-WE-Çayönü*), as well as the 230 ancient Eurasian individuals published by Mathieson et al. [15] (*HO-WE-230*). We further executed a Pandora stability analysis on the global subset of the HO SNP array (*HO-GLOB*) and the SNP Panels 1 – 13 of the Affymetrix Human Origins array [19]. Since all these datasets only contain human genotype data, we also analyzed datasets comprising animal genotypes. We used the dataset of Sudanese goat genotypes published by Rahmatalla et al. [23] (*Goats*) and also used Southeast-European sheep genotype data published by Ciani et al. [5] (*Sheep*). Note that while we made a substantial effort to correctly reproduce the HO-based datasets, we did not invest the same effort to obtain or re-assemble other published datasets we use. The reason for this is that we found the task of merely reassembling datasets to be highly challenging for the publications we considered. This is mainly because the instructions provided regarding data sources and preprocessing routines lack the level of detail required to correctly recreate the analyzed datasets. In other cases, while more detailed descriptions are available, datasets comprise a vast number of (hand-picked) individuals obtained from previously published datasets. For instance, Changmai et al. [4] use individuals obtained from 19 distinct publications. To limit the time invested for reassembling datasets, we decided to not include such datasets here. In addition to these empirical datasets, we also analyzed 16 simulated genotype datasets published by Yi and Latch [29] that exhibit varying proportions of missing data.

In the following, we only show and discuss the results of the Pandora stability estimates for PCA analyses of the empirical datasets in greater detail and summarize the key findings of the same analyses using MDS based on the F_{ST} population distance. We further summarize the PCA and MDS results of the Pandora analyses of the 16 simulated datasets, as well as an exemplary sliding-window analysis. The detailed results of the MDS, simulated datasets, and sliding-window analyses are in Supplementary Material Section 2. Note that we provide all scripts to reproduce the dataset assemblies, as well as results, on GitHub `https://github.com/tschuelia/PandoraPaper`. We additionally provide further information on the software versions and hardware we used in Supplementary Material Section 3.

3.1 Stability of Empirical Datasets

3.1.1 PCA Stability Estimation

For the PCA analyses of the empirical datasets, we used the default *smartpca* settings, that is, 10 principal components and five rounds of outlier detection. For datasets with ancient individuals, we initially computed the PCA using the modern individuals only and subsequently projected the ancient individuals (using the `lsqproject: YES` option in *smartpca*) onto them, prior to conducting the stability estimation. Supplementary Table S1 shows the detailed results of the Pandora uncertainty estimation for all 19 analyzed empirical datasets.

Overall, all 19 empirical datasets are relatively stable under bootstrapping ($PS \geq 0.73$). We do, however, observe individuals with low PSVs in *all* datasets. The most uncertain and unstable dataset is *Panel 7* of the Affymetrix Human Origins array. Its PS is 0.73 with an average PSV of 0.67. We observe the highest stability for the *HO-GLOB* dataset, with a PS of 0.98 and average PSV of 0.98 as well. Figure 2 shows the empirical cumulative distribution functions (eCDF) for the PSVs for *Panel 7* (left) and *HO-GLOB* (right) visualizing the stability differences in terms of PSVs for the two datasets.

While there is a slight trend for higher PSVs with higher numbers of SNPs, this correlation is not statistically significant (Pearson correlation $r = 0.48$, p-value $= 0.051$). Our results suggest that one should always consider both, the PS, and PSVs to estimate the stability of a dataset. For example, the *Goats* dataset appears to be very stable with a PS of 0.92. However, estimating the stability of the individual samples unravels a lower stability, with an average PSV of only 0.75. In contrast, for the *Panel 4* dataset the PSVs suggest high stability averaging to 0.93, but the overall PS yields a more conservative stability estimate of 0.83.

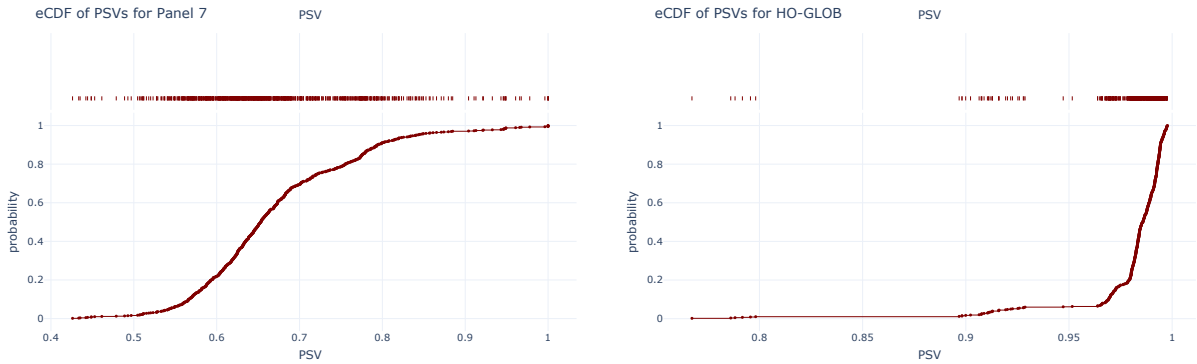


Figure 2: ECDF of PSVs for the *Panel 7* (left) and the *HO-GLOB* (right) datasets.

3.1.2 MDS Stability Estimation

In addition to the PCA analyses, we performed Pandora stability analyses using MDS, the alternative dimensionality reduction technique available in Pandora. Supplementary Table S3 shows the resulting PS estimates, as well as PSV summary statistics. We used the *smartpca* F_{ST} distance metric between populations as input for the MDS analysis and reduced the data to two dimensions. Overall, MDS analyses appear to be more stable than PCA analyses. This observation holds true for both the PS, and the PSVs. In our experiments, we only observed three exceptions: *Panel 5* ($PS_{PCA} = 0.89$ vs. $PS_{MDS} = 0.86$), *Panel 9* ($PS_{PCA} = 0.92$ vs. $PS_{MDS} = 0.85$), and *Goats* ($PS_{PCA} = 0.92$ vs. $PS_{MDS} = 0.81$).

3.1.3 Analysis of *HO-WE* Data

In the following, we analyze the stability of *HO-WE* and *HO-WE-Çayönü* in greater detail to showcase the use of Pandora, and to demonstrate the utility of Pandora for population genetic and ancient DNA research.

We first examine the *HO-WE* dataset comprising modern West-Eurasian individuals, focusing on the effects of *smartpca*'s outlier detection and "shrinkmode" on the stability and the support values. In the original publication of the Human Origins dataset, Lazaridis et al. [14] set the number of outlier iterations in *smartpca* to zero. According to Pandora, the stability of the *HO-WE* dataset *without* outlier detection is 0.95. Interestingly, we observed that performing PCA with five outlier iterations (the *smartpca* default setting) resulted in a slightly lower stability than when performing the stability analysis using zero outlier iterations during the *smartpca* PCA computation (PS = 0.89). We further observed a substantial impact of *smartpca*'s "shrinkmode" on the stability analyses. According to the *smartpca* documentation, individuals from the same underlying population will appear as different populations in the resulting PCA, if only some of these individuals are used to compute the principal components. The shrinkmode attempts to address this problem. Since its introduction in EIGENSOFT version 7.2.0, this shrinkmode is frequently used in PCA analyses. For example, Altınışık et al. [1] use the default five outlier iterations and the shrinkmode for analyzing the *HO-WE-Çayönü* dataset. To assess the impact of the shrinkmode, we analyzed the *HO-WE* dataset, once with shrinking enabled, and once with shrinking disabled. In both cases, we set the number of outlier iterations to *smartpca's* default setting of five and performed the Pandora stability estimation using 100 bootstrap replicates. While the PS for *HO-WE* without shrinking is 0.89, it decreases to 0.66 with shrinking enabled. While we expected that using shrinking would increase stability, the contrary was the case, as the PS decreased substantially. We observe a similar effect for the PSVs: with shrinking enabled,

the PSVs tend to be lower than with shrinkmode disabled. This effect is visualized in Figure 3 showing the eCDF for the PSVs under both settings. Using shrinking yields low support for two Italian ($PSV = 0.5$) and 21 Middle-Eastern individuals ($PSV \leq 0.62$; see Figure 4b). Conversely, we observe the lowest support with shrinking disabled for five individuals from the Balkan peninsula (Bulgarian, Greek, Turkish; $0.60 \leq PSV \leq 0.62$; see Figure 4a). In both cases, with shrinking enabled and disabled, we observe a general trend for decreased stability for Balkan and Middle-Eastern individuals.

For an in-depth analysis of the *HO-WE-Çayönü* dataset, we focus on the uncertainty regarding the population assignment(s) of ancient individuals. We use *smartpca* with five rounds of outlier detection and shrinking enabled according to Altınışık et al. [1]. Note that for the results stated in Supplementary Table S1, we analyzed the *HO-WE-Çayönü* dataset with shrinking disabled. Consequently, the PS and PSVs in the following analyzes will deviate from those in Supplementary Table S1. Using the *smartpca* option `lsqproject: YES`, we computed all PCAs using the modern individuals only and then projected the 13 ancient Çayönü individuals onto the resulting embeddings. Figure 5 plots the result of the PCA analysis on the unbootstrapped dataset. The gray dots indicate the modern individuals, while the colored stars indicate the projected ancient individuals.

The overall PS of the *HO-WE-Çayönü* dataset is 0.66 with an average PSV of 0.85. The PSVs of the ancient Çayönü individuals range from 0.49 (*cay033*) to 0.81 (*cay007*). In their analyses, Altınışık et al. [1] computed a 95% confidence interval using the ellipse function in *smartpca*. This confidence interval indicates a high projection uncertainty, especially for the *cay015* individual. Our results are in line with these findings, as the *cay015* individual has a PSV of only 0.54. Figure 6 visualizes this behavior. The figure shows the computed PCA embeddings of two randomly selected bootstrap replicate datasets. Prior to plotting, we applied Procrustes transformation to match the two embeddings as accurately as possible (hence the shape of the PCA deviates from the shape shown in Figure 5). The dots indicate the projection of individuals from the first bootstrap, the stars indicate the projection of individuals from the second bootstrap. The three highlighted individuals are the *HO-WE-Çayönü* individuals with the lowest PSVs (*cay033*: 0.49, *cay014*, *cay015*: 0.54). We can clearly see that the position of the individuals in the PCA spaces of both embeddings differ substantially, hence the low PSVs. For individuals with such low PSVs, conclusions based on PCA analyses should be drawn with extreme caution. For example, individual *cay033* with the lowest PSV is projected closest to Palestinian individuals in the PCA of the unbootstrapped *HO-WE-Çayönü* dataset. However, when considering the two randomly selected bootstraps we used in Figure 6, *cay033* is projected closest to Syrian individuals (bootstrap replicate #1) or Libyan Jew individuals (bootstrap replicate #2). These observations are again in line with the findings of Altınışık et al. [1], as they conclude that Çayönü is a genetically diverse population.

3.2 Stability of Simulated Datasets

We further estimated the stability of 16 simulated datasets with varying fractions of missing data (1%, 10%, 20%) published by Yi and Latch [29]. Supplementary Table S8 shows the stability estimates using PCA. Note that due to the relatively small number of simulated SNPs (5,000), we only compute and compare two principal components. Yi and Latch [29] conclude that PCA is more unstable the more data is declared as missing in a dataset. Our results confirmed this trend, as we observed lower PS estimates when more data was missing. Supplementary Table S9 shows the results of stability estimates using the F_{ST} population distance and subsequent MDS dimensionality reduction (again reducing the data to two dimensions). Since all simulated datasets only comprise three distinct populations, the stability of each dataset with MDS is almost 1.0 for all datasets. Consequently, we additionally performed MDS analyses using a Hamming distance computed between pairs of individuals that corrects for missing data (see Supplementary Table S10). This distance metric corresponds to the `--distance 'flat-missing'` option in PLINK 1.9 [3]. Using this alternative distance met-

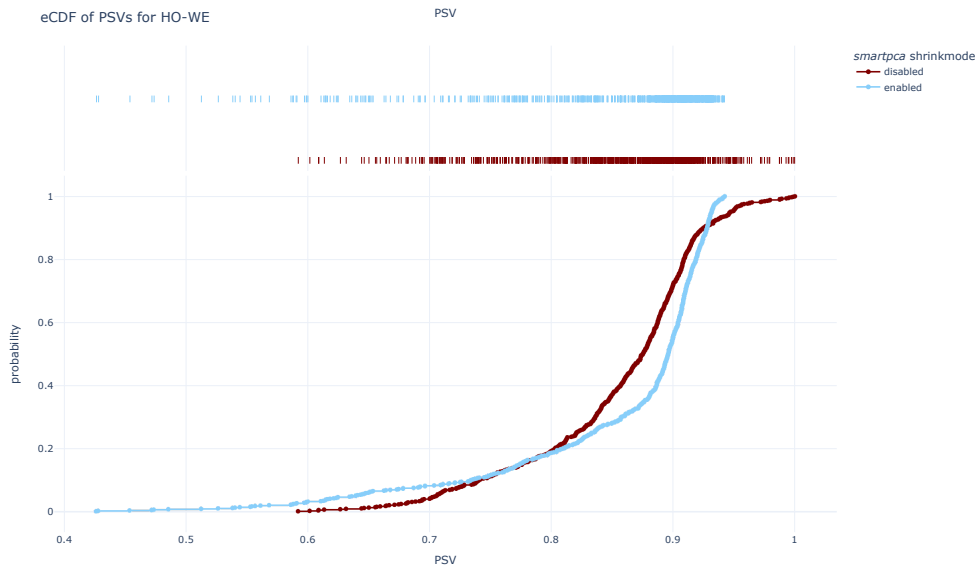
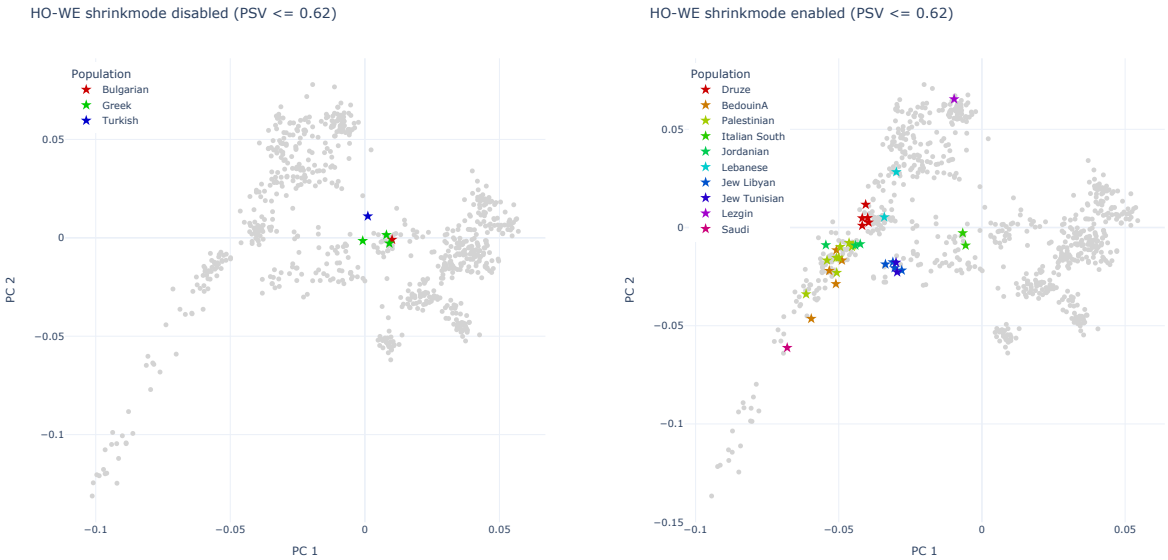


Figure 3: ECDF of PSVs for all individuals of the *HO-WE* dataset for Pandora analyses with shrinkmode disabled (dark red traces) and shrinkmode enabled (light blue traces).

ric, the simulated datasets are substantially less stable compared to using the F_{ST} population distance, but also compared to PCA analyses. We thus advise users of Pandora to choose the appropriate distance metric, as well as dimensionality reduction technique, carefully based on the requirements of subsequent downstream analyses and prior knowledge about the data under study.

3.3 Speedup and Accuracy under the Bootstrap Convergence Criterion

In the following, we analyze the influence of the implemented bootstrap convergence check as described in Section 2.4 on the stability estimates and the runtime of Pandora. For each of the empirical datasets, we performed the following analyses. As baseline, we executed Pandora with the full set of 100 bootstrap replicates (default Pandora setting) and disabled the bootstrap convergence check. We denote the runtime of this execution by T_{100}, the PS as PS_{100}, and the PSVs as PSV_{100}. We further executed Pandora twice with the convergence check enabled, but with a different convergence tolerance in each execution. In one execution, we set the convergence tolerance to 0.05 (5%) and in a second execution to a more conservative value of 0.01 (1%). We denote the results of both executions as $T_{5\%}, PS_{5\%}, PSV_{5\%}$ and $T_{1\%}, PS_{1\%}, PSV_{1\%}$, respectively. We further report the number of bootstraps required for convergence for both executions ($N_{5\%}$, $N_{1\%}$). For the baseline execution, we set $N_{100} = 100$ (Pandora default setting) and the number of threads to 20. To compare the runtimes, we computed the speedups $S_{5\%} = \frac{T_{100}}{T_{5\%}}$ and $S_{1\%} = \frac{T_{100}}{T_{1\%}}$. A speedup below 1 indicates that the runtime with the convergence check enabled exceeds the runtime of computing all 100 replicates without checking for convergence. In cases where the bootstrap procedure does not converge before all 100 bootstraps are computed, the speedup is necessarily below 1 due to the additional convergence checks. To analyze the potential loss of accuracy of the early termination on the resulting stability analyses, we



(a) *HO-WE* individuals with a $PSV \leq 0.62$ for Pandora analyses *without* the *smartpca* shrinkmode.

(b) *HO-WE* individuals with a $PSV \leq 0.62$ for Pandora analyses *with* the *smartpca* shrinkmode.

Figure 4: Visualization of the first two principal components of the unbootstrapped *HO-WE* dataset for one Pandora analysis without shrinking (left figure) and one Pandora analysis with shrinking (right figure). Both figures show the computed embedding as gray dots, and individuals with a $PSV \leq 0.62$ are highlighted. Colors indicate populations, as stated in the legends.

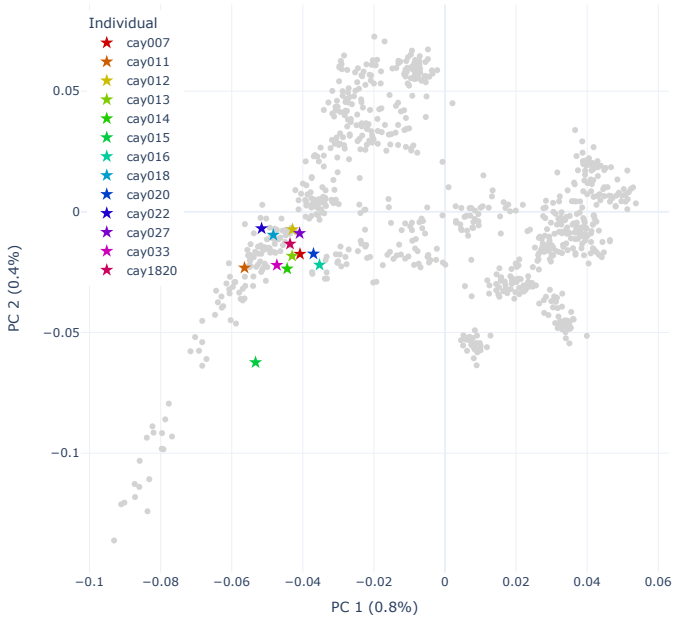


Figure 5: Projection of ancient Çayönü individuals onto a PCA of the unbootstrapped modern individuals of the *HO-WE-Çayönü* dataset. The gray dots indicate the modern individuals and the colored stars indicate the projected ancient individuals.

computed the relative deviation of PS values

$$\frac{|PS_{5\%} - PS_{100}|}{PS_{100}}$$

and

$$\frac{|PS_{1\%} - PS_{100}|}{PS_{100}},$$

as well as the relative deviation of PSVs

$$\frac{|PSV^i_{5\%} - PSV^i_{100}|}{PSV^i_{100}};\ i = 1, \dots M$$

and

$$\frac{|PSV^i_{1\%} - PSV^i_{100}|}{PSV^i_{100}};\ i = 1, \dots M.$$

Figure 7 shows the results for the empirical datasets using PCA embedding. Aggregated over all datasets, we visualize the distribution of the relative deviation of PSVs (Figure 7a) and the distribution of speedups (Figure 7b) via box plots, at a 5% and 1% convergence tolerance. At 5%, we observe a speedup of 2.46 with almost no loss of accuracy in terms of PS deviation (0% for 12 out of 19 empirical datasets and 1.1% for the remaining 7 datasets). We do however observe PSV deviations of up to 16.5% for a single individual. However, this notable deviation appears to concern only a few samples, as we observe PSV deviations only up to 2.8% when averaged on a per-dataset basis. At 1%, we record an average speedup of only 1.17 with no PS deviation on all datasets and PSV deviations up to 6.8% for a single individual or up to 1.4% when averaged on a per-dataset basis.

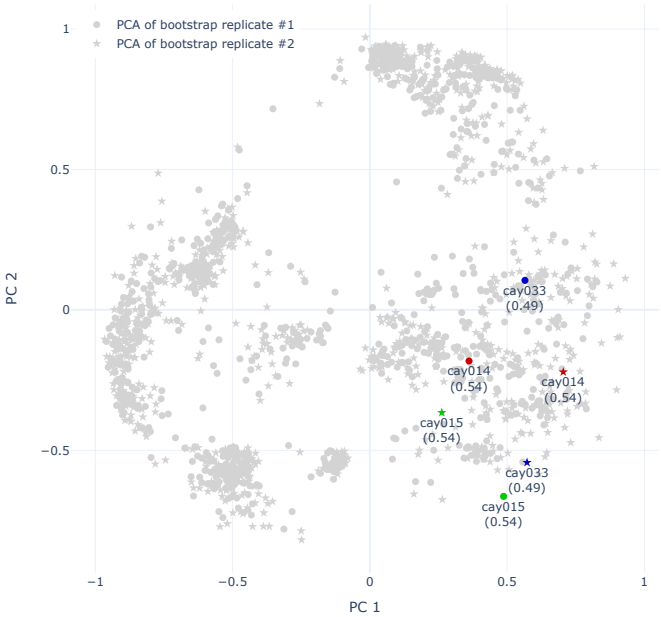


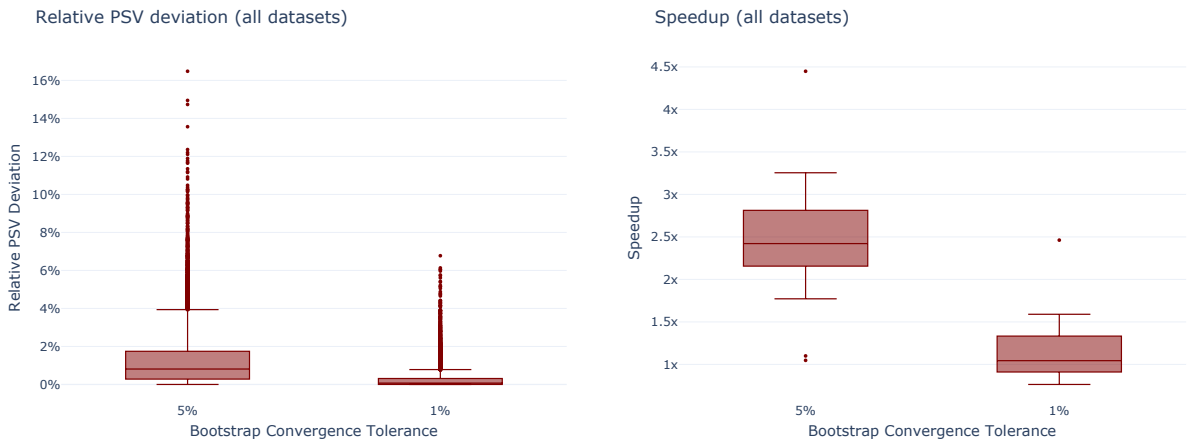
Figure 6: The three Çayönü individuals with the lowest PSVs plotted for two randomly selected bootstrap PCA results. The gray dots indicate the projections from one bootstrap, the gray stars indicate the projections from the second bootstrap. The highlighted individuals indicate the respective projection of the three Çayönü individuals in both PCAs.

For more details, we refer to Supplementary Table S2 showing speedups, PS deviations, maximal and average PSV deviations, by dataset and convergence tolerance. Supplementary Figure S3 additionally shows box plots of PSV deviations by dataset, and Supplementary Table S6 the absolute runtimes of Pandora PCA analyses.

For PCA analyses, we conclude that the bootstrap convergence heuristic provides a good trade-off between the accuracy of stability estimates and runtime, unless high accuracy of PSVs is paramount.

Finally, choosing a suitable convergence tolerance yields a trade-off between accuracy and runtime. We recommend using Pandora's default setting of 5%, if the stability analysis focuses on the overall PS. If one is more interested in the PSVs of specific individuals, we recommend decreasing the convergence tolerance to 1%. In both cases, we observed an average speedup above 1 compared to simply computing 100 bootstrap replicates. However, as stated above, for datasets indicating no convergence before the full set of 100 bootstrap replicates was computed, the speedup is below 1 due to the computational overhead of the convergence check. In our experiments, this was only the case for analyses with a 1% convergence tolerance for seven datasets. We observed the highest runtime overhead for *Panel 3* with a 25% runtime increase.

The results for the bootstrap convergence check for MDS embeddings are shown in Supplementary Material Section 2.1.4. Interestingly, even for the less conservative (default) convergence tolerance of 5%, we observe a speedup below 1 despite early stop-



(a) PSV deviations for all individuals of all empirical datasets for both convergence tolerance settings.

(b) Speedup per dataset for both convergence tolerance settings.

Figure 7: PSV deviations and speedups for both analyzed convergence tolerance settings. The box plots show the data for all empirical datasets using PCA analyses.

ping after 20 replicates on most datasets. Compared to PCA, performing MDS analyses is fast (see Supplementary Section 2.1.4). Hence, the runtime overhead introduced by the repeated application of Procrustes Transformations during the bootstrap convergence check exceeds the runtime of the relatively fast MDS computation. Consequently, we advise users to disable the bootstrap convergence check for Pandora MDS analyses.

The number of threads determines the frequency of the convergence checks. As a consequence, the number of computed bootstraps, as well as the runtime and resulting stability estimates, may depend on the number of threads. For the results in Supplementary Table S3, we executed Pandora with 20 threads. Consequently, Pandora checked for convergence every 20 bootstrap replicates. To determine the impact of this thread-based heuristic, we reran the above analyses (using a 5% convergence tolerance and PCA embeddings) using only 10 cores, thus performing the convergence check twice as often as with 20 cores. Table S5 in the Supplementary Material lists the number of computed replicates (N_{10}, N_{20}), the PSs (PS_{10}, PS_{20}) and their relative deviation $(\frac{|PS_{10}-PS_{20}|}{PS_{20}})$ for both Pandora settings. On average, the PS deviation is 0.4%, with values ranging from 0% to 1.12%. We conclude that the number of threads provided for the Pandora execution does not substantially influence the resulting stability estimates.

3.4 Influence of k on the PCS

Running Pandora on various population genetics datasets, we observed a substantial influence of the number of clusters k used for k-means clustering on the PCS cluster similarity score. Figure 8 depicts this observation for the *HO-WE* dataset with 100 bootstrap replicates for $k \in [3, 15]$. While we do observe a general trend for a lower PCS with a higher number of clusters k, we also observe fluctuations for $k \in [5, 8]$ without a clear trend.

We thus strongly recommend providing a reasonable k to Pandora based on prior knowledge or preliminary experiments. We further note that the PCS should only be reported in conjunction with the number of clusters k. As we do not have the prior knowledge for determining a reasonable or optimal number

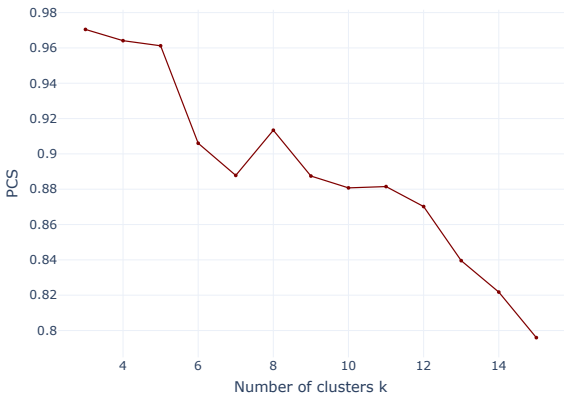


Figure 8: PCS for the *HO-WE* dataset as a function of the number of clusters k used for k-means clustering.

of clusters for all datasets in our study, we only report the PCS values for the HO-WE dataset and decided against reporting corresponding PCS values for the remaining datasets we used.

3.5 Sliding-Window Analysis

In addition to the standard bootstrapping stability analysis, Pandora provides a sliding-window stability estimation specifically targeted at whole genome data. The idea is to quantify the stability of a genotype dataset under dimensionality reduction when using different subsets of the whole genome data for the analyses. We performed a sliding-window analysis using genotypes of the 1000 Genomes project [8]. We used the autosomal genotype data of the 929 individuals collected in the Human Genotype Diversity Project (HGDP) [2]. The 929 individuals of the HGDP dataset belong to 57 global populations. We LD-pruned the dataset prior to the Pandora analyses using *PLINK 2.0* [3] with a window size of 50, stride 5, and $r^2 = 0.5$. The final dataset contained 7,005,100 SNPs.

In the subsequent Pandora sliding-window analysis, we set the number of windows to compute to 12, the number of principal components to 10, and we used the default settings of *smartpca*. The Pandora stability for the HGDP dataset with 12 windows is 0.74 and the PSVs range from 0.6 to 1 with an average of 0.86. We observed the least stability for West-African samples (Mandinka and Yoruba). Supplementary Figures S8 and S9 depict the first two principal components for all 12 windows. Interestingly, the Pandora stability increases to $PS = 0.77$ when splitting the dataset into 50 windows. However, the individuals' stabilities decrease, with a minimum PSV of 0.48 and an average PSV of 0.78. With this analysis, we observed the lowest PSVs for Western- and South-Asian samples (Sindhi, Brahui, Balochi, Makrani).

4 Discussion and Conclusion

Pandora estimates the stability of a genotype dataset under dimensionality reduction using bootstrapping. In addition to an overall stability estimate (PS), Pandora applies k-means clustering to all bootstrapped embeddings to estimate the stability of the assigned clusters across all bootstrap replicates (PCS). Furthermore, Pandora computes a bootstrap support value for each individual in the genotype dataset (PSVs). The latter is particularly useful for datasets with projections of ancient individuals onto an embedding computed on modern individuals only, as the PSVs for the ancient individuals provide a confidence value regarding their placement in the embedding space relative to modern individuals. The lower the support for an individual, the more careful users should be about drawing conclusions about that individual based on dimensionality reduction analyses.

Pandora supports PCA and MDS analyses, and provides additional functionality for sliding-window-based stability estimation of dimensionality reduction along whole genome data. Compared to a single PCA or MDS analysis, Pandora introduces a substantial runtime and resource overhead. Our bootstrap convergence assessment partially alleviates this and reduces the overhead to a reasonable amount.

Our exemplary analyses demonstrate the utility of the Pandora stability analysis as it helps to prevent potentially erroneous and hence misleading conclu-

sions based on unstable dimensionality reduction. We believe that Pandora will be of high value to population genetic studies as the inherent uncertainty of PCA and MDS analyses can now be seamlessly and routinely assessed via an easy-to-use tool and thereby contribute to avoid potentially biased conclusions.

Pandora bootstraps the SNPs and subsequently reduces the dimensionality of the dataset along the same dimension of the dataset, that is, the SNPs. Pandora does not implement a resampling of the individuals (*jackknifing*), as Elhaik [7] demonstrated the sensitivity of PCA studies to (subjective) selection of individuals. The author showed that PCA analyses can be manipulated to confirm the desired hypothesis, depending on the selection of individuals used for PCA analyses. Consequently, for Pandora, we refrained from implementing a jackknifing procedure and thus Pandora results can not directly be used to justify a subjective selection of individuals. However, a jackknifing procedure could allow to estimate the bias induced by specific individuals and in conjunction with the stability estimates of Pandora, especially the PSVs, contribute to identify an objective, meaningful set of individuals for population genetics studies. Devising such a selection criterion is the subject of future work.

5 Acknowledgements

We thank N. Ezgi Altınışık for the very helpful discussion of Pandora, as well as the helpful insights in the methodology of population genetics studies. We thank the aDNA lab at IMBB-FORTH in Heraklion, Crete for the useful initial discussions and ideas. We further thank Jonas Haag for the invaluable help in debugging Python multiprocessing. This work was financially supported by the Klaus Tschira Foundation and by the European Union (EU) under Grant Agreement No 101087081 (Comp-Biodiv-GR).

Funded by the European Union

References

[1] N. E. Altınışık, D. D. Kazancı, A. Aydoğan, H. C. Gemici, Ömür Dilek Erdal, S. Sarıaltun, K. B. Vural, D. Koptekin, K. Gürün, E. Sağlıcan, et al. A genomic snapshot of demographic and cultural dynamism in upper mesopotamia during the neolithic transition. *Science Advances*, 8(44):eabo3609, 2022. doi: 10.1126/sciadv.abo3609.

[2] A. Bergström, S. A. McCarthy, R. Hui, M. A. Almarri, Q. Ayub, P. Danecek, Y. Chen, S. Felkel, P. Hallast, J. Kamm, H. Blanché, J.-F. Deleuze, H. Cann, S. Mallick, D. Reich, M. S. Sandhu, P. Skoglund, A. Scally, Y. Xue, R. Durbin, and C. Tyler-Smith. Insights into human genetic variation and population history from 929 diverse genomes. *Science*, 367(6484):eaay5012, 2020. doi: 10.1126/science.aay5012.

[3] C. C. Chang, C. C. Chow, L. C. Tellier, S. Vattikuti, S. M. Purcell, and J. J. Lee. Second-generation plink: rising to the challenge of larger and richer datasets. *GigaScience*, 4 (1):s13742–015–0047–8, 2015. doi: 10.1186/s13742-015-0047-8.

[4] P. Changmai, K. Jaisamut, J. Kampuansai, W. Kutanan, N. E. Altınışık, O. Flegontova, A. Inta, E. Yüncü, W. Boonthai, H. Pamjav, D. Reich, and P. Flegontov. Indian genetic heritage in southeast asian populations. *PLOS Genetics*, 18(2):1–25, 02 2022. doi: 10.1371/journal.pgen.1010036.

[5] E. Ciani, S. Mastrangelo, A. Da Silva, F. Marroni, M. Ferenčaković, P. Ajmone-Marsan, H. Baird, M. Barbato, L. Colli, C. Delvento, et al. On the origin of european sheep as revealed by the diversity of the balkan breeds and by optimizing population-genetic analysis tools. *Genetics Selection Evolution*, 52(1), 2020. doi: 10.1186/s12711-020-00545-7.

[6] B. Efron. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7(1):1 – 26, 1979. doi: 10.1214/aos/1176344552.

[7] E. Elhaik. Principal component analyses (pca)-based findings in population genetic studies are highly biased and must be reevaluated. *Scientific Reports*, 12, 08 2022. doi: 10.1038/s41598-022-14395-4.

[8] S. Fairley, E. Lowy-Gallego, E. Perry, and P. Flicek. The International Genome Sample Resource (IGSR) collection of open human genomic variation resources. *Nucleic Acids Research*, 48(D1):D941–D947, 2019. doi: 10.1093/nar/gkz836.

[9] J. Felsenstein. Confidence limits on phylogenies: An approach using the bootstrap. *Evolution*, 39(4):783–791, 1985. doi: 10.1111/j.1558-5646.1985.tb00420.x.

[10] A. Fisher, B. Caffo, B. Schwartz, and V. Zipunnikov. Fast, exact bootstrap principal component analysis for $p > 1$ million. *Journal of the American Statistical Association*, 111(514):846–860, 2016. doi: 10.1080/01621459.2015.1062383.

[11] E. B. Fowlkes and C. L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):553–569, 1983.

[12] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, et al. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020. doi: 10.1038/s41586-020-2649-2.

[13] J. R. Hughey, P. Paschou, P. Drineas, D. Mastropaolo, D. M. Lotakis, P. A. Navas, M. Michalodimitrakis, J. A. Stamatoyannopoulos, and G. Stamatoyannopoulos. A european population in minoan bronze age crete. *Nature Communications*, 4(1):1861, 2013. doi: 10.1038/ncomms2871.

[14] I. Lazaridis, N. Patterson, A. Mittnik, G. Renaud, S. Mallick, K. Kirsanow, P. H. Sudmant, J. G. Schraiber, S. Castellano, M. Lipson, et al. Ancient human genomes suggest three ancestral populations for present-day europeans. *Nature*, 513(7518):409–413, 2014. doi: 10.1038/nature13673.

[15] I. Mathieson, I. Lazaridis, N. Rohland, S. Mallick, N. Patterson, S. A. Roodenberg, E. Harney, K. Stewardson, D. Fernandes, M. Novak, et al. Genome-wide patterns of selection in 230 ancient eurasians. *Nature*, 528(7583):499–503, 2015. doi: 10.1038/nature16152.

[16] P. Menozzi, A. Piazza, and L. Cavalli-Sforza. Synthetic maps of human gene frequencies in europeans. *Science*, 201(4358):786–792, 1978. doi: 10.1126/science.356262.

[17] A. Miles, Pyup.Io Bot, R. Murillo, P. Ralph, J. Kelleher, M. Schelker, R. Pisupati, S. Rae, and T. Millar. GitHub: cggh/scikit-allel v1.3.7, 2023.

[18] N. Patterson, A. L. Price, and D. Reich. Population structure and eigenanalysis. *PLOS Genetics*, 2(12):1–20, 12 2006. doi: 10.1371/journal.pgen.0020190.

[19] N. Patterson, P. Moorjani, Y. Luo, S. Mallick, N. Rohland, Y. Zhan, T. Genschoreck, T. Webster, and D. Reich. Ancient Admixture in Human History. *Genetics*, 192(3):1065–1093, 2012. doi: 10.1534/genetics.112.145037.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[21] A. Price, N. Patterson, R. Plenge, M. Weinblatt, N. Shadick, and D. Reich. Principal components analysis corrects for stratification in genome-wide association studies. *Nature genetics*, 38:904–9, 09 2006. doi: 10.1038/ng1847.

[22] S. Purcell, B. Neale, K. Todd-Brown, L. Thomas, M. A. Ferreira, D. Bender, J. Maller, P. Sklar, P. I. de Bakker, M. J.

Daly, and P. C. Sham. Plink: A tool set for whole-genome association and population-based linkage analyses. *The American Journal of Human Genetics*, 81(3):559–575, 2007. doi: 10.1086/519795.

[23] S. A. Rahmatalla, D. Arends, M. Reissmann, A. Said Ahmed, K. Wimmers, H. Reyer, and G. A. Brockmann. Whole genome population genetics analysis of sudanese goats identifies regions harboring genes associated with major traits. *BMC Genetics*, 18(1), 2017. doi: 10.1186/s12863-017-0553-z.

[24] G. Schwarz. Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461 – 464, 1978. doi: 10.1214/aos/1176344136.

[25] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[26] C. Wang, Z. A. Szpiech, J. H. Degnan, M. Jakobsson, T. J. Pemberton, J. A. Hardy, A. B. Singleton, and N. A. Rosenberg. Comparing spatial maps of human population-genetic variation using procrustes analysis. *Statistical Applications in Genetics and Molecular Biology*, 9(1), 2010. doi: 10.2202/1544-6115.1493.

[27] S. WRIGHT. The genetical structure of populations. *Annals of Eugenics*, 15(1):323–354, 1949. doi: 10.1111/j.1469-1809.1949.tb02451.x.

[28] X. Yi and E. K. Latch. Nonrandom missing data can bias principal component analysis inference of population genetic structure. *Molecular Ecology Resources*, 22(2):602–611, 2022. doi: 10.1111/1755-0998.13498.

[29] X. Yi and E. K. Latch. Nonrandom missing data can bias principal component analysis inference of population genetic structure. *Molecular Ecology Resources*, 22(2):602–611, 2022. doi: 10.1111/1755-0998.13498.